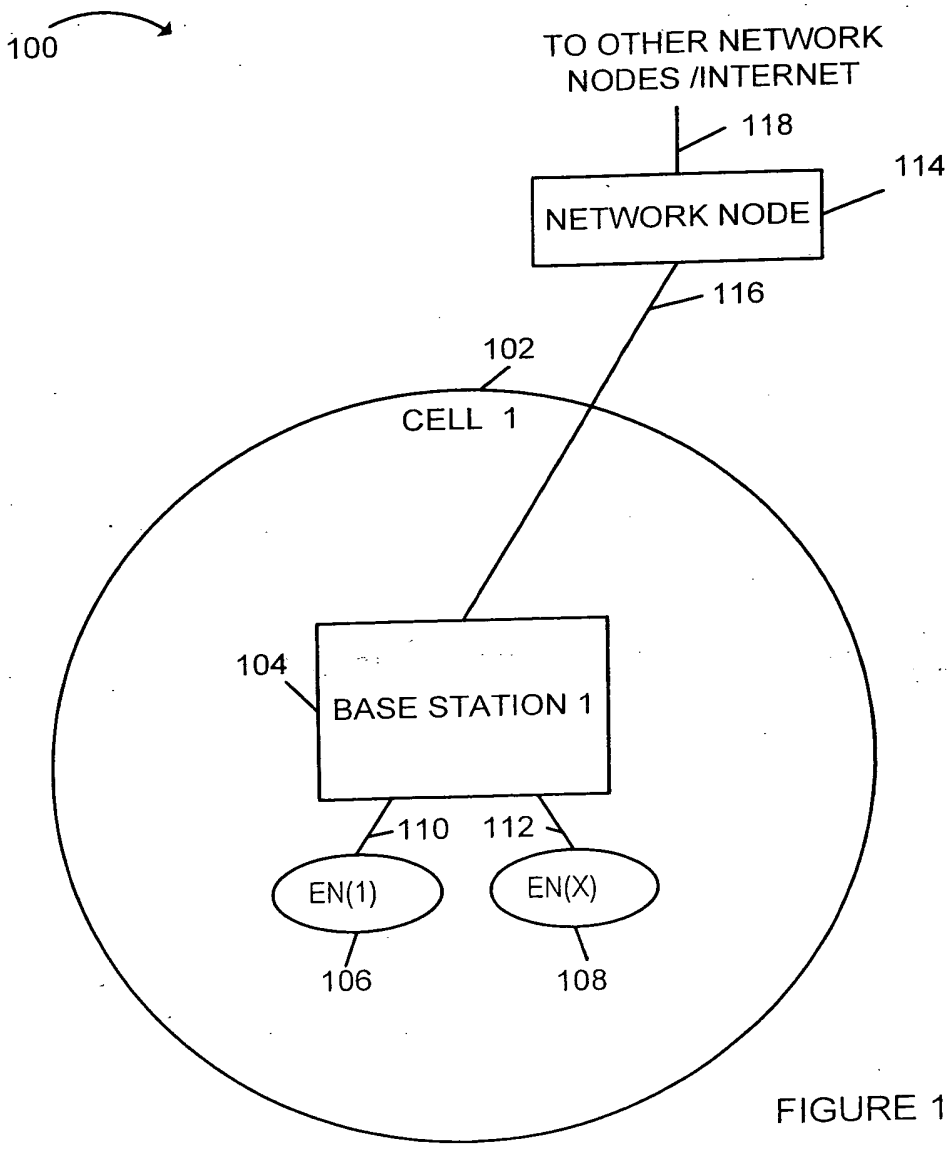

100
TO OTHER NETWORK NODES /INTERNET
118
114
NETWORK NODE
116
102
CELL 1
104
BASE STATION 1
110
112
EN(1)
EN(X)
106
108

FIGURE 1

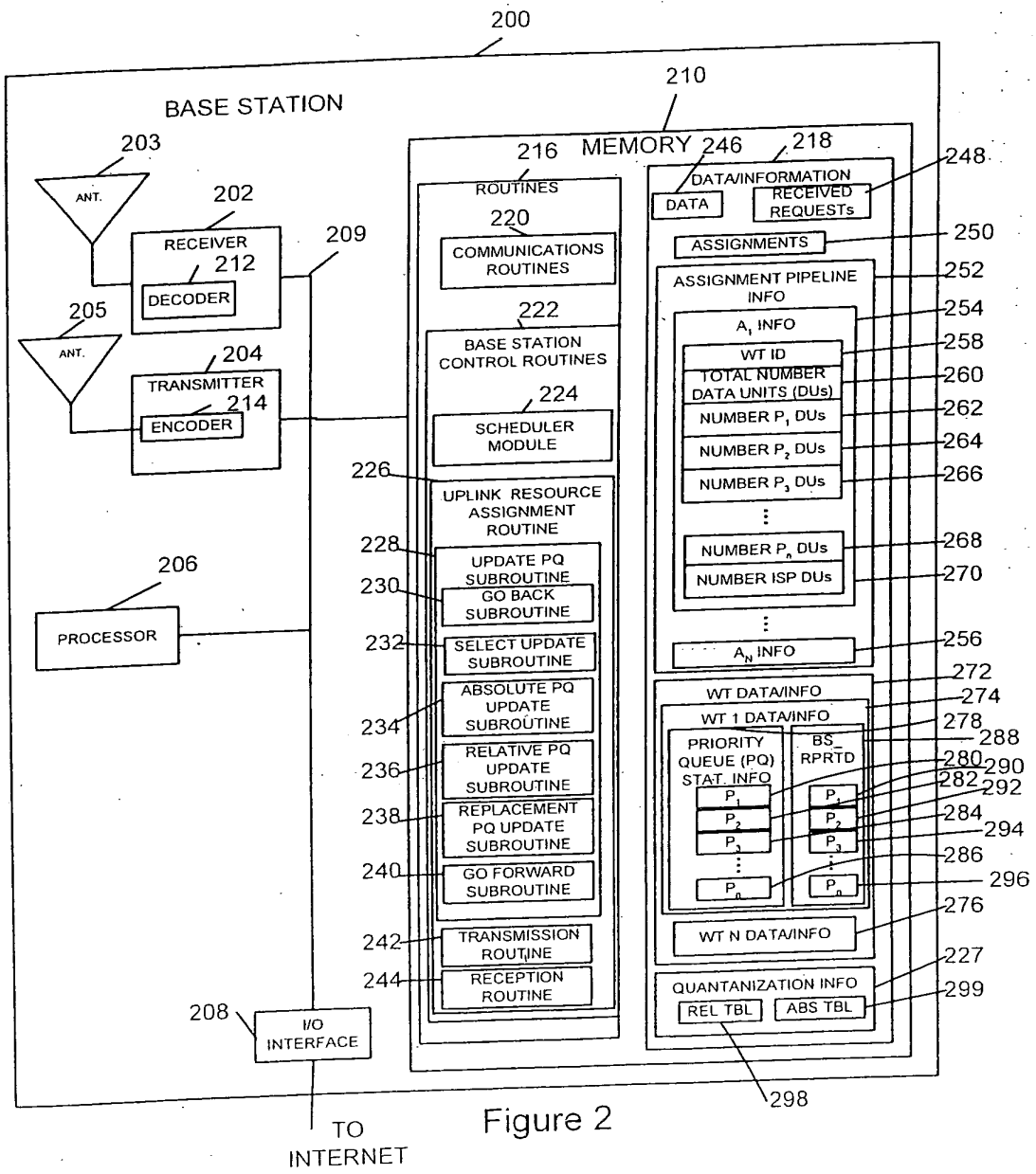

200
BASE STATION
210
MEMORY
216
246
218
248
203
ANT.
202
RECEIVER
212
DECODER
209
205
ANT.
204
TRANSMITTER
214
ENCODER
206
PROCESSOR
208
I/O INTERFACE
TO INTERNET
ROUTINES
220
COMMUNICATIONS ROUTINES
222
BASE STATION CONTROL ROUTINES
224
SCHEDULER MODULE
226
UPLINK RESOURCE ASSIGNMENT ROUTINE
228
UPDATE PQ SUBROUTINE
230
GO BACK SUBROUTINE
232
SELECT UPDATE SUBROUTINE
234
ABSOLUTE PQ UPDATE SUBROUTINE
236
RELATIVE PQ UPDATE SUBROUTINE
238
REPLACEMENT PQ UPDATE SUBROUTINE
240
GO FORWARD SUBROUTINE
242
TRANSMISSION ROUTINE
244
RECEPTION ROUTINE
DATA/INFORMATION
DATA
RECEIVED REQUESTs
ASSIGNMENTS
250
ASSIGNMENT PIPELINE INFO
252
A1 INFO
254
WT ID
258
TOTAL NUMBER DATA UNITS (DUs)
260
NUMBER P1 DUs
262
NUMBER P2 DUs
264
NUMBER P3 DUs
266
NUMBER Pn DUs
268
NUMBER ISP DUs
270
AN INFO
256
WT DATA/INFO
272
WT 1 DATA/INFO
274
PRIORITY QUEUE (PQ) STAT. INFO
278
BS_ RPRTD
288
280
290
282
292
284
294
286
296
WT N DATA/INFO
276
QUANTANIZATION INFO
227
REL TBL
ABS TBL
299
298

Figure 2

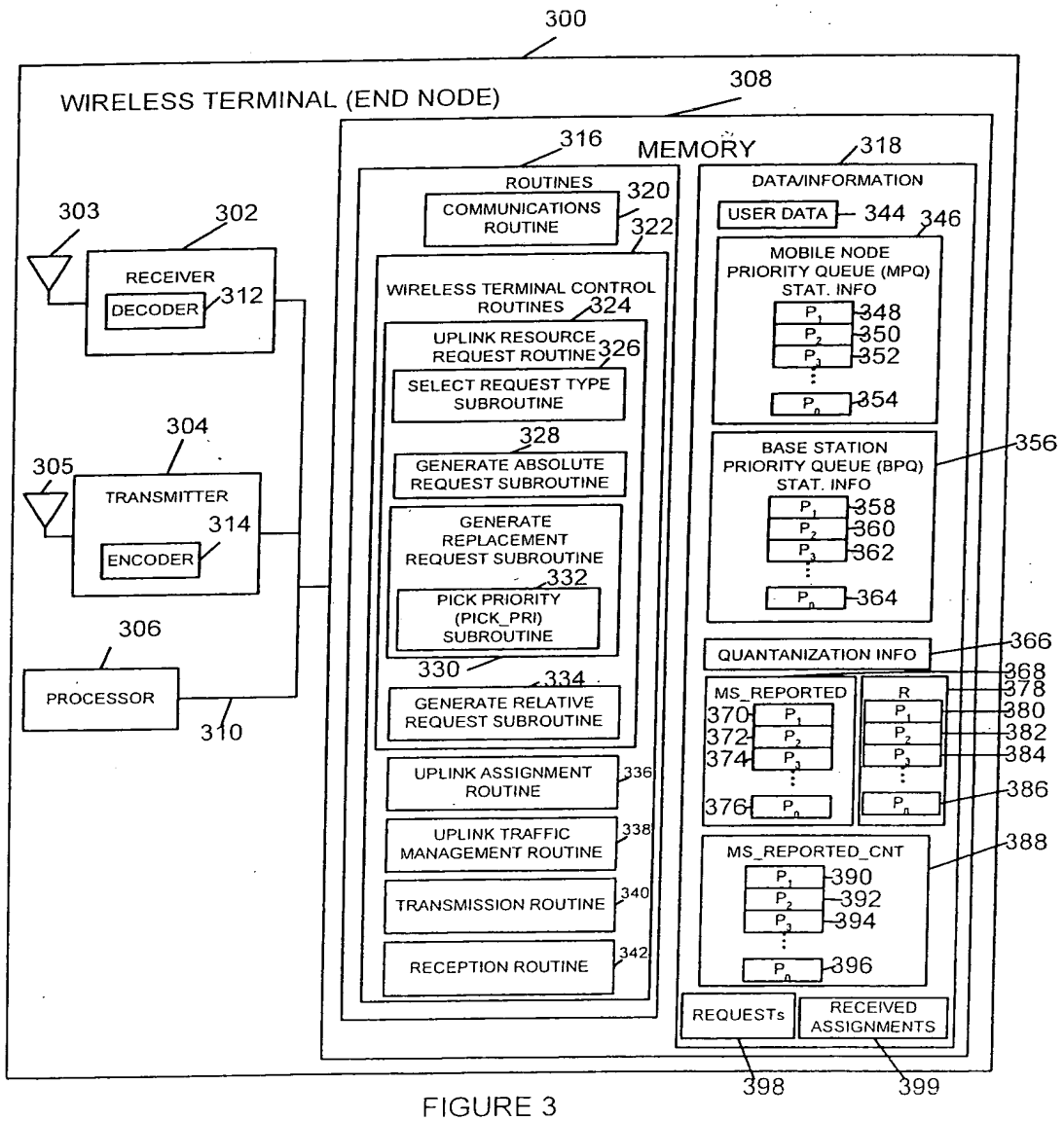

300
WIRELESS TERMINAL (END NODE)
308
MEMORY
316
318
ROUTINES
320
COMMUNICATIONS ROUTINE
322
WIRELESS TERMINAL CONTROL ROUTINES
324
UPLINK RESOURCE REQUEST ROUTINE
326
SELECT REQUEST TYPE SUBROUTINE
328
GENERATE ABSOLUTE REQUEST SUBROUTINE
GENERATE REPLACEMENT REQUEST SUBROUTINE
332
PICK PRIORITY (PICK_PRI) SUBROUTINE
330
334
GENERATE RELATIVE REQUEST SUBROUTINE
UPLINK ASSIGNMENT ROUTINE
336
UPLINK TRAFFIC MANAGEMENT ROUTINE
338
TRANSMISSION ROUTINE
340
RECEPTION ROUTINE
342
DATA/INFORMATION
USER DATA
344
346
MOBILE NODE PRIORITY QUEUE (MPQ) STAT. INFO
348
350
352
354
BASE STATION PRIORITY QUEUE (BPQ) STAT. INFO
356
358
360
362
364
QUANTANIZATION INFO
366
368
MS_REPORTED
370
372
374
376
R
378
380
382
384
386
MS_REPORTED_CNT
388
390
392
394
396
REQUESTs
RECEIVED ASSIGNMENTS
398
399
303
302
RECEIVER
DECODER
312
304
305
TRANSMITTER
ENCODER
314
306
PROCESSOR
310

FIGURE 3

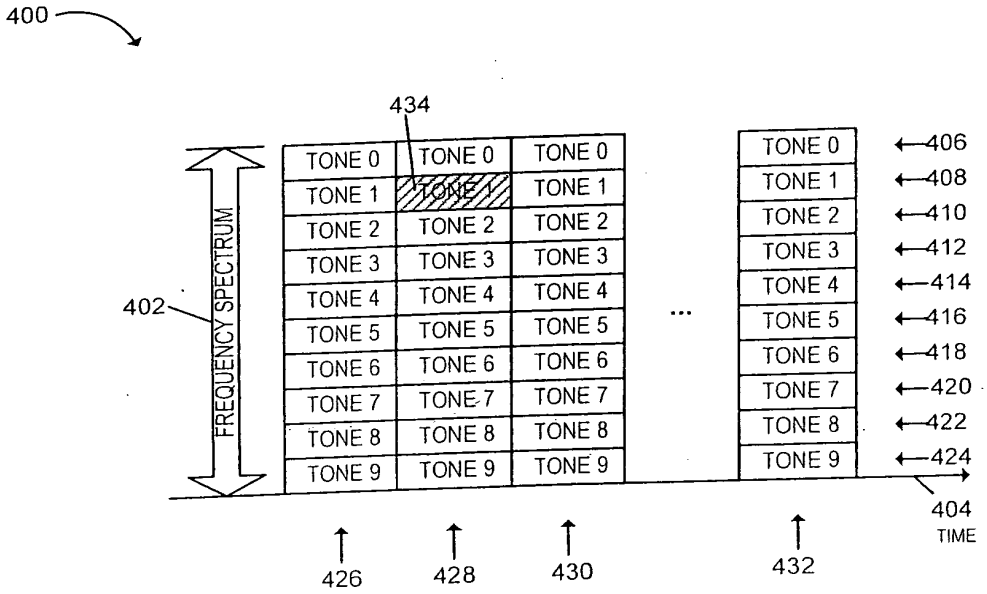

400
434
402
FREQUENCY SPECTRUM
TONE 0
TONE 1
TONE 2
TONE 3
TONE 4
TONE 5
TONE 6
TONE 7
TONE 8
TONE 9
406
408
410
412
414
416
418
420
422
424
404
TIME
426
428
430
432

FIGURE 4

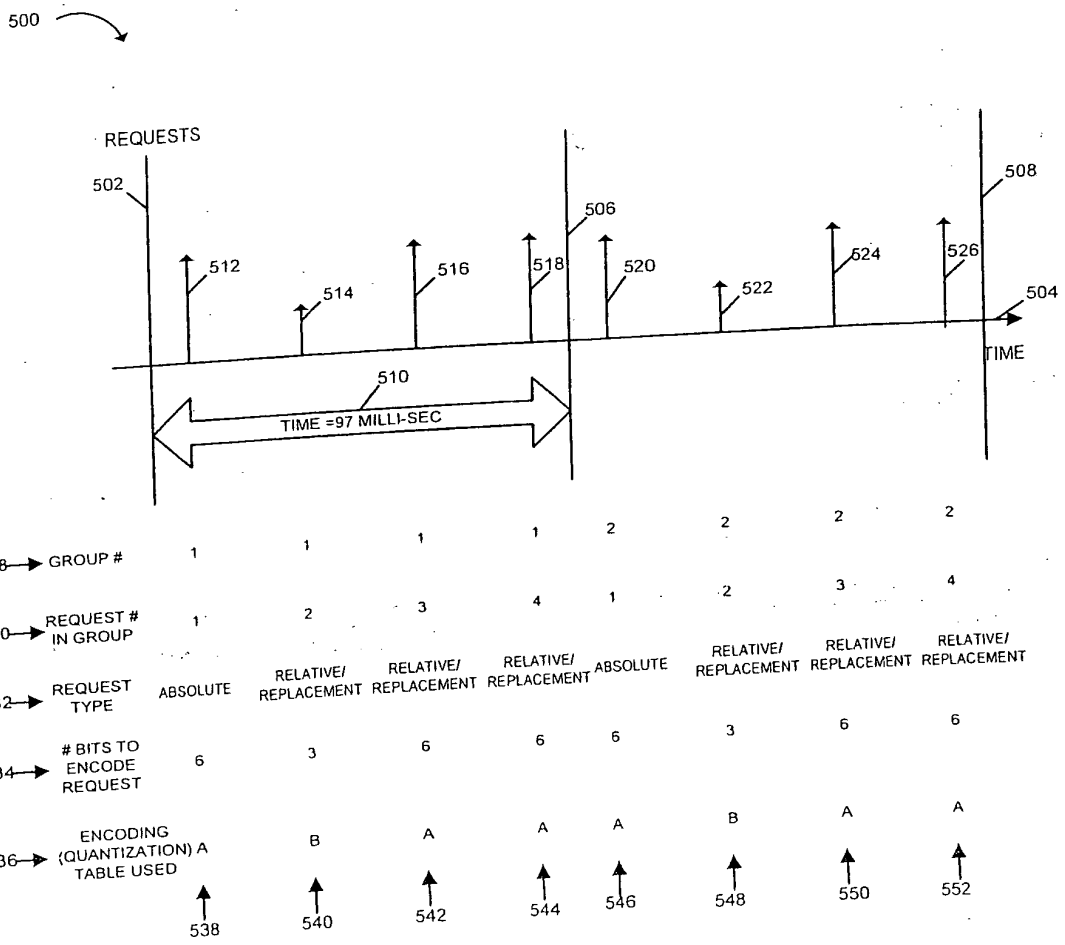
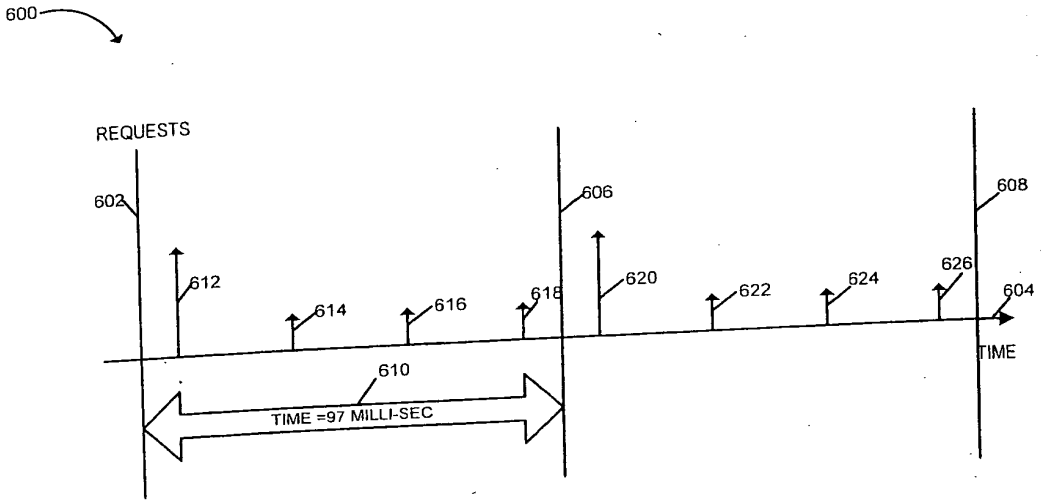

500
REQUESTS
502
504
506
508
510
512
514
516
518
520
522
524
526
TIME
TIME =97 MILLI-SEC
528 GROUP #
530 REQUEST # IN GROUP
532 REQUEST TYPE
534 # BITS TO ENCODE REQUEST
536 ENCODING (QUANTIZATION) TABLE USED
1 1 1 1 2 2 2 2
1 2 3 4 1 2 3 4
ABSOLUTE
RELATIVE/ REPLACEMENT
RELATIVE/ REPLACEMENT
RELATIVE/ REPLACEMENT
ABSOLUTE
RELATIVE/ REPLACEMENT
RELATIVE/ REPLACEMENT
RELATIVE/ REPLACEMENT
6 3 6 6 6 3 6 6
A B A A A B A A
538 540 542 544 546 548 550 552

Fig. 5



628 GROUP #	1	1	1	1	2	2	2	2	
630 REQUEST # IN GROUP	1	2	3	4	1	2	3	4	
632 REQUEST TYPE	ABSOLUTE	RELATIVE/ REPLACEMENT	RELATIVE/ REPLACEMENT	RELATIVE/ REPLACEMENT	ABSOLUTE	RELATIVE/ REPLACEMENT	RELATIVE/ REPLACEMENT	RELATIVE/ REPLACEMENT	
634 # BITS TO ENCODE REQUEST	6	2	2	2	6	2	2	2	
636 ENCODING (QUANTIZATION) A TABLE USED		C	C	C	A	C	C	C	
	638	640	642	644	646	648	650	652	

Fig. 6

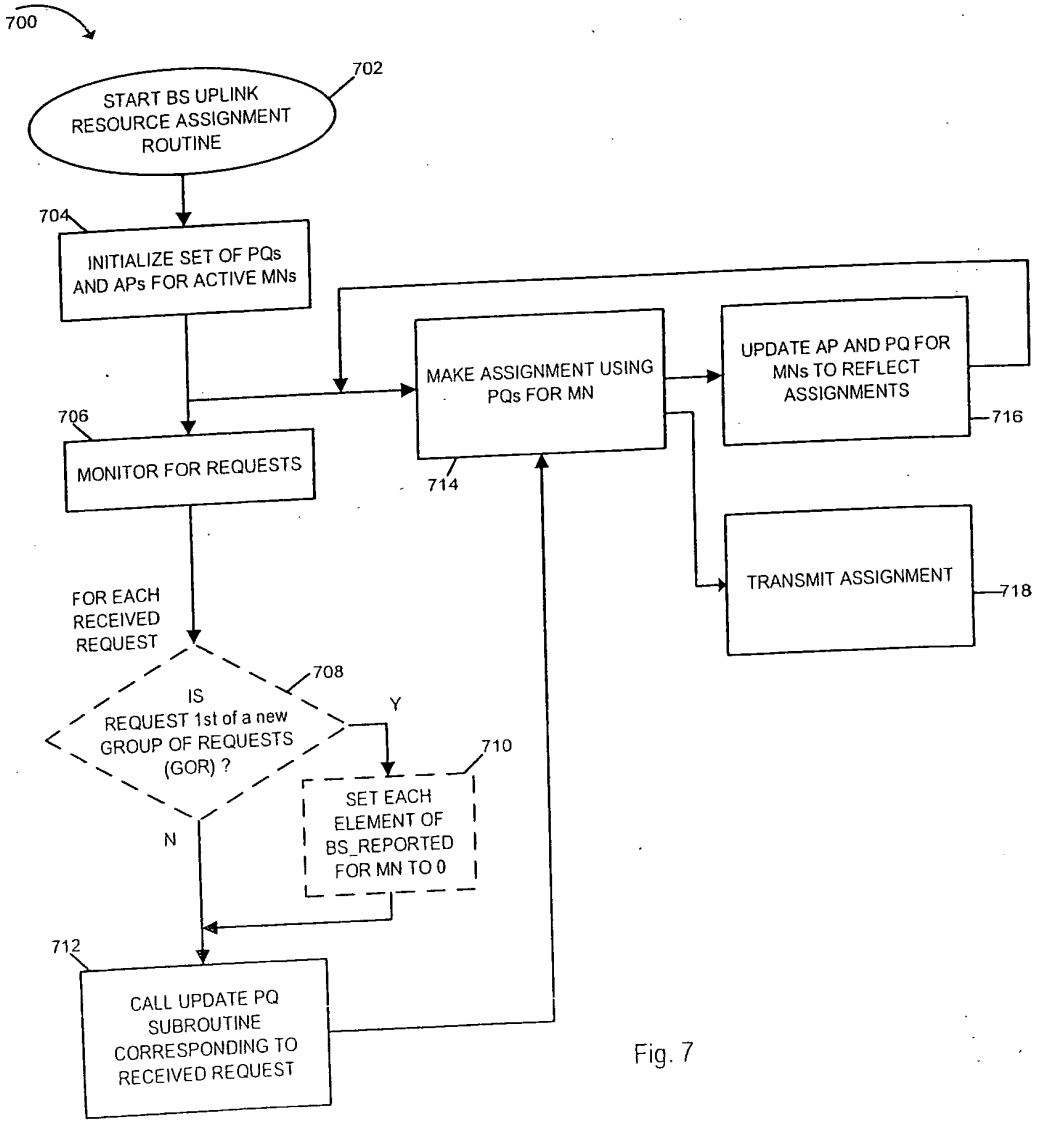

700
START BS UPLINK RESOURCE ASSIGNMENT ROUTINE
702
INITIALIZE SET OF PQs AND APs FOR ACTIVE MNs
704
MONITOR FOR REQUESTS
706
FOR EACH RECEIVED REQUEST
IS REQUEST 1st of a new GROUP OF REQUESTS (GOR) ?
708
Y
N
SET EACH ELEMENT OF BS_REPORTED FOR MN TO 0
710
CALL UPDATE PQ SUBROUTINE CORRESPONDING TO RECEIVED REQUEST
712
MAKE ASSIGNMENT USING PQs FOR MN
714
UPDATE AP AND PQ FOR MNs TO REFLECT ASSIGNMENTS
716
TRANSMIT ASSIGNMENT
718

Fig. 7

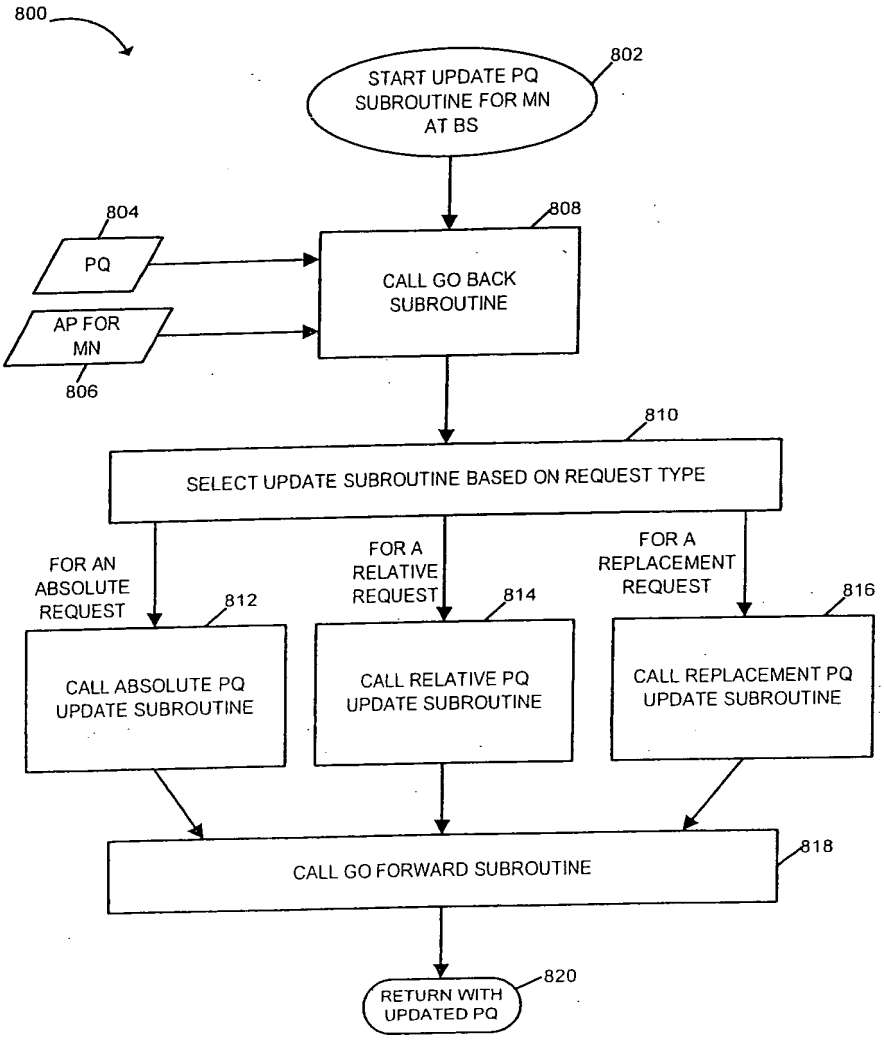

800
START UPDATE PQ SUBROUTINE FOR MN AT BS
802
804
PQ
AP FOR MN
806
808
CALL GO BACK SUBROUTINE
810
SELECT UPDATE SUBROUTINE BASED ON REQUEST TYPE
FOR AN ABSOLUTE REQUEST
FOR A RELATIVE REQUEST
FOR A REPLACEMENT REQUEST
812
814
816
CALL ABSOLUTE PQ UPDATE SUBROUTINE
CALL RELATIVE PQ UPDATE SUBROUTINE
CALL REPLACEMENT PQ UPDATE SUBROUTINE
818
CALL GO FORWARD SUBROUTINE
820
RETURN WITH UPDATED PQ

Fig. 8

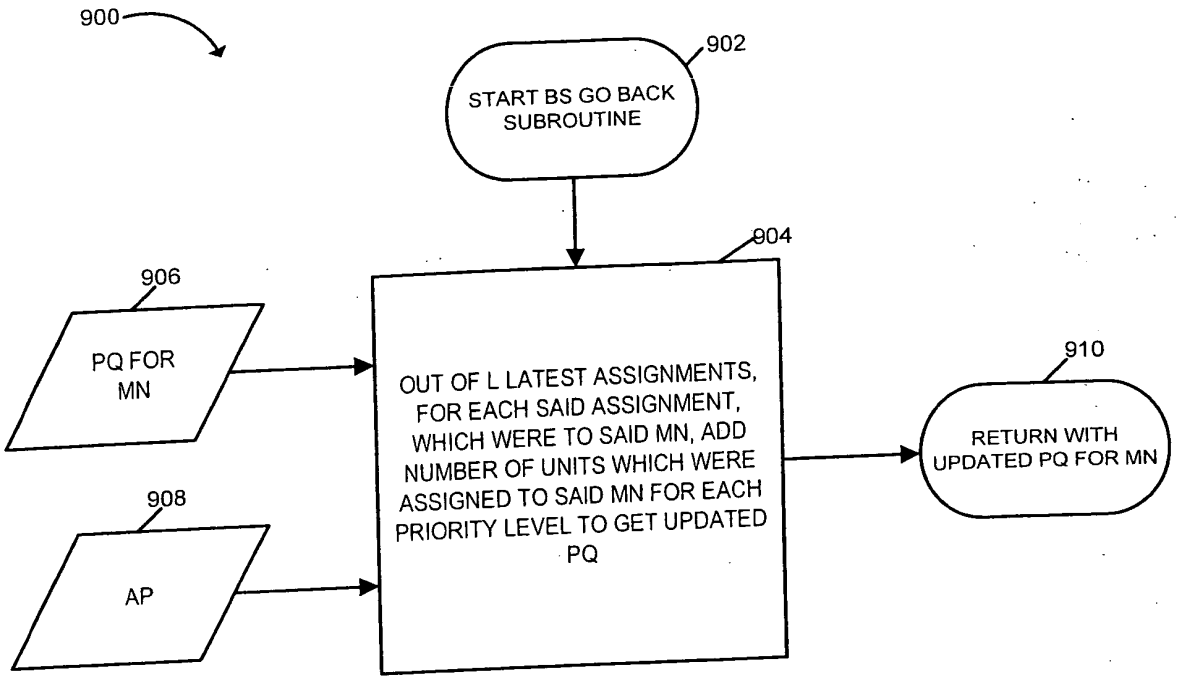

900
902
START BS GO BACK SUBROUTINE
904
906
PQ FOR MN
908
AP
OUT OF L LATEST ASSIGNMENTS, FOR EACH SAID ASSIGNMENT, WHICH WERE TO SAID MN, ADD NUMBER OF UNITS WHICH WERE ASSIGNED TO SAID MN FOR EACH PRIORITY LEVEL TO GET UPDATED PQ
910
RETURN WITH UPDATED PQ FOR MN

Fig. 9

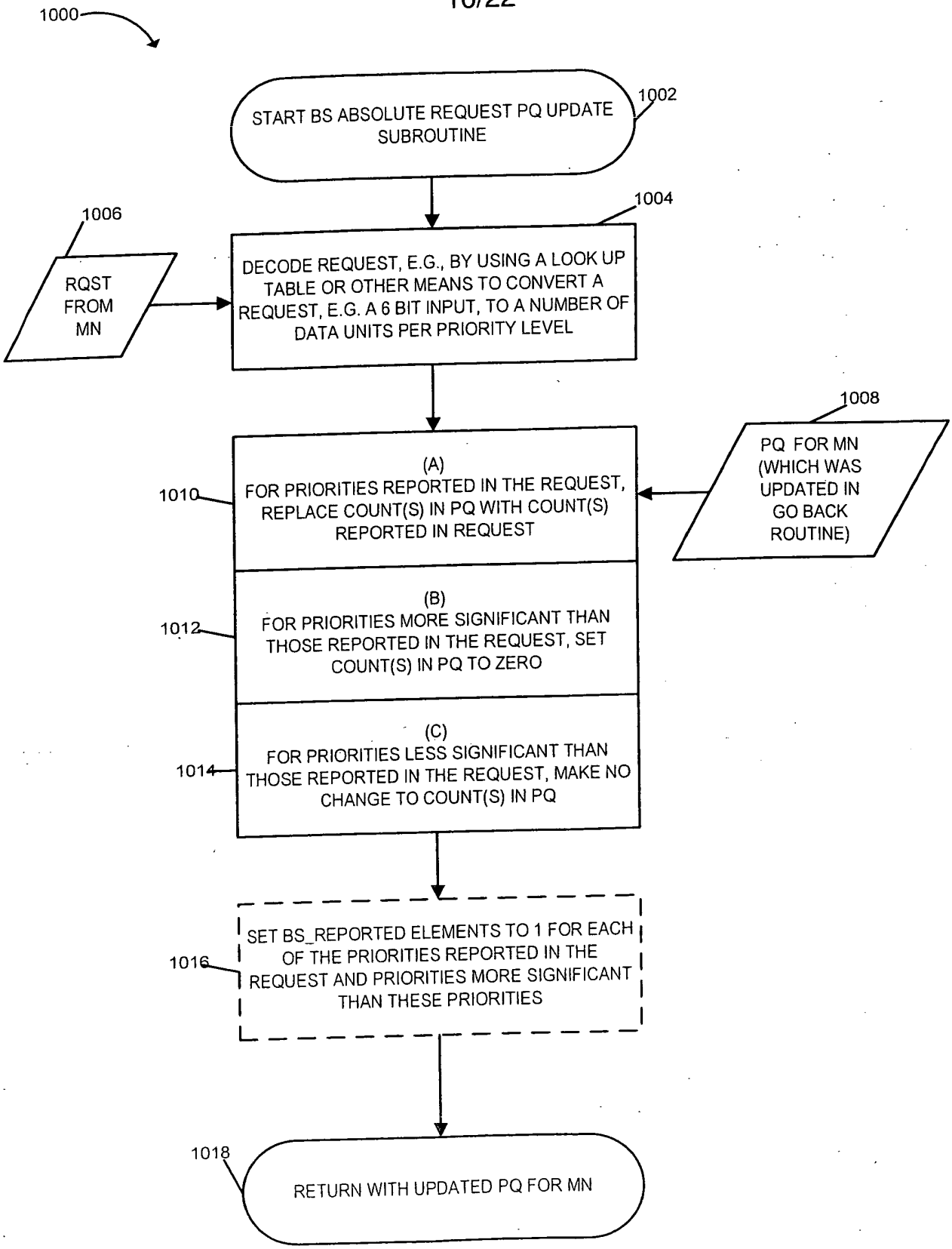

1000
START BS ABSOLUTE REQUEST PQ UPDATE SUBROUTINE
1002
1004
1006
RQST FROM MN
DECODE REQUEST, E.G., BY USING A LOOK UP TABLE OR OTHER MEANS TO CONVERT A REQUEST, E.G. A 6 BIT INPUT, TO A NUMBER OF DATA UNITS PER PRIORITY LEVEL
1008
PQ FOR MN (WHICH WAS UPDATED IN GO BACK ROUTINE)
1010
(A) FOR PRIORITIES REPORTED IN THE REQUEST, REPLACE COUNT(S) IN PQ WITH COUNT(S) REPORTED IN REQUEST
1012
(B) FOR PRIORITIES MORE SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST, SET COUNT(S) IN PQ TO ZERO
1014
(C) FOR PRIORITIES LESS SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST, MAKE NO CHANGE TO COUNT(S) IN PQ
1016
SET BS_REPORTED ELEMENTS TO 1 FOR EACH OF THE PRIORITIES REPORTED IN THE REQUEST AND PRIORITIES MORE SIGNIFICANT THAN THESE PRIORITIES
1018
RETURN WITH UPDATED PQ FOR MN

Fig. 10

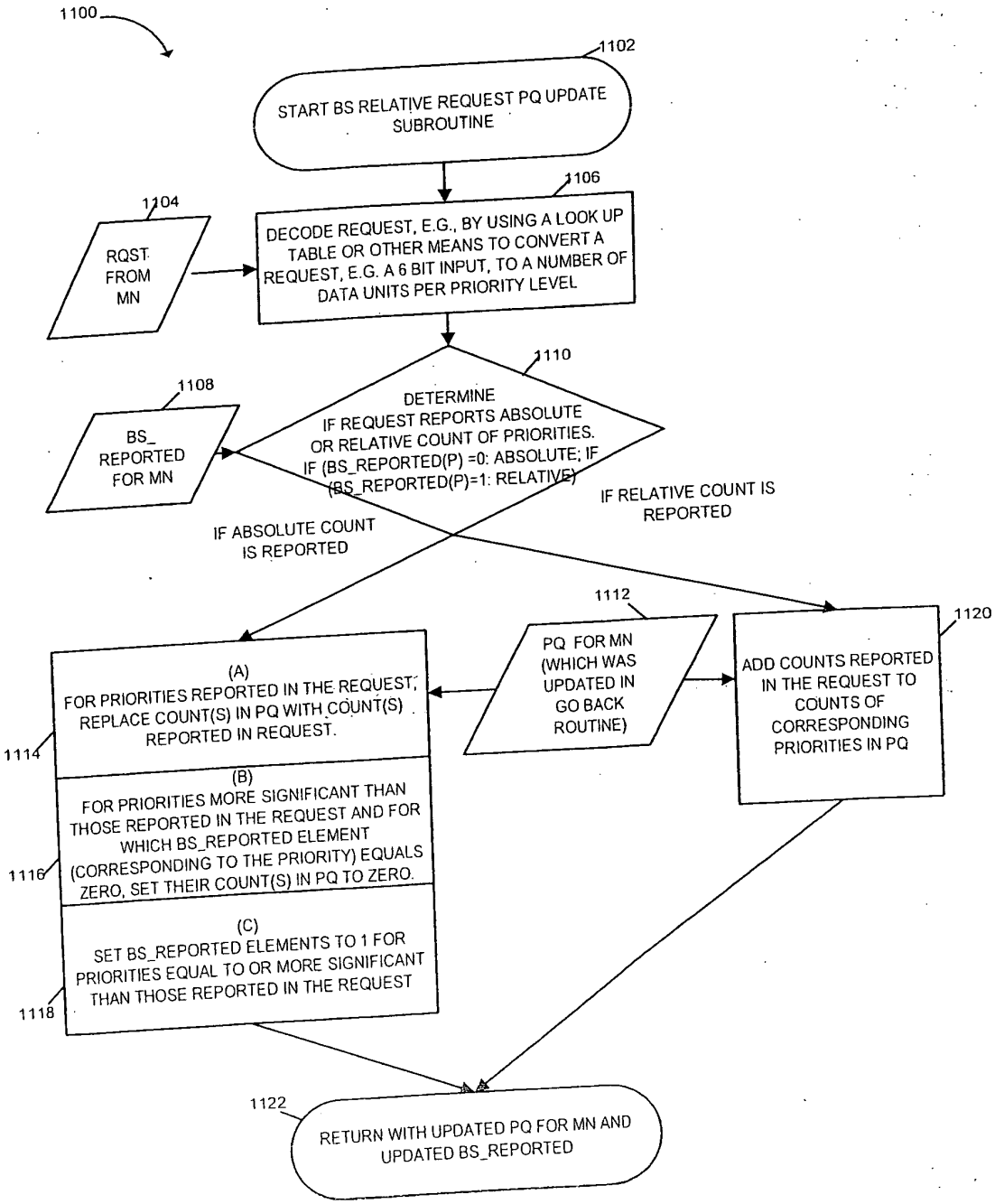

1100
1102
START BS RELATIVE REQUEST PQ UPDATE SUBROUTINE
1104
RQST FROM MN
1106
DECODE REQUEST, E.G., BY USING A LOOK UP TABLE OR OTHER MEANS TO CONVERT A REQUEST, E.G. A 6 BIT INPUT, TO A NUMBER OF DATA UNITS PER PRIORITY LEVEL
1108
BS_ REPORTED FOR MN
1110
DETERMINE IF REQUEST REPORTS ABSOLUTE OR RELATIVE COUNT OF PRIORITIES. IF (BS_REPORTED(P) =0: ABSOLUTE; IF (BS_REPORTED(P)=1: RELATIVE)
IF ABSOLUTE COUNT IS REPORTED
IF RELATIVE COUNT IS REPORTED
1112
PQ FOR MN (WHICH WAS UPDATED IN GO BACK ROUTINE)
1120
ADD COUNTS REPORTED IN THE REQUEST TO COUNTS OF CORRESPONDING PRIORITIES IN PQ
1114
(A) FOR PRIORITIES REPORTED IN THE REQUEST, REPLACE COUNT(S) IN PQ WITH COUNT(S) REPORTED IN REQUEST.
1116
(B) FOR PRIORITIES MORE SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST AND FOR WHICH BS_REPORTED ELEMENT (CORRESPONDING TO THE PRIORITY) EQUALS ZERO, SET THEIR COUNT(S) IN PQ TO ZERO.
1118
(C) SET BS_REPORTED ELEMENTS TO 1 FOR PRIORITIES EQUAL TO OR MORE SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST
1122
RETURN WITH UPDATED PQ FOR MN AND UPDATED BS_REPORTED

Fig. 11

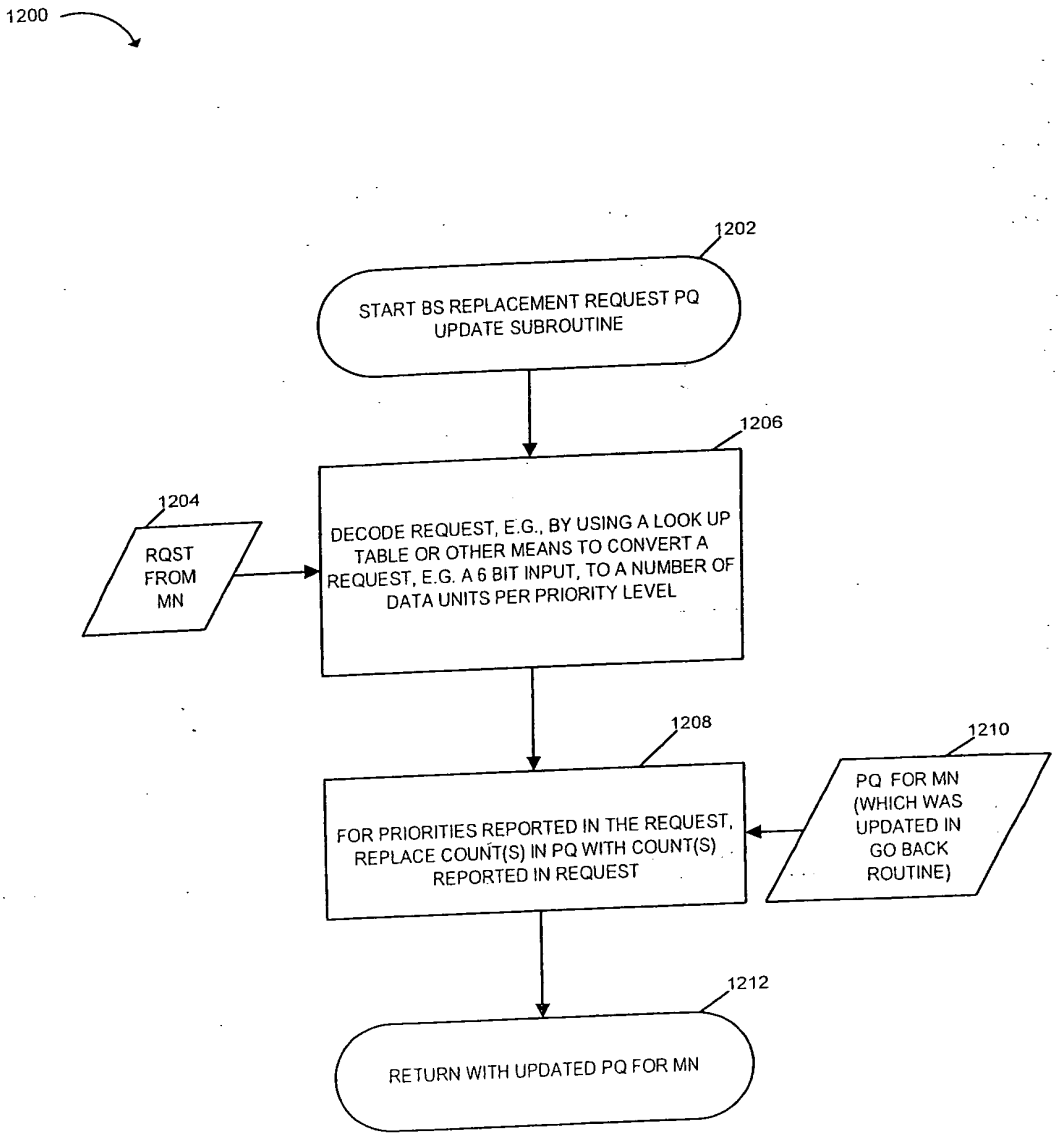

1200
1202
START BS REPLACEMENT REQUEST PQ UPDATE SUBROUTINE
1206
1204
RQST FROM MN
DECODE REQUEST, E.G., BY USING A LOOK UP TABLE OR OTHER MEANS TO CONVERT A REQUEST, E.G. A 6 BIT INPUT, TO A NUMBER OF DATA UNITS PER PRIORITY LEVEL
1208
1210
FOR PRIORITIES REPORTED IN THE REQUEST, REPLACE COUNT(S) IN PQ WITH COUNT(S) REPORTED IN REQUEST
PQ FOR MN (WHICH WAS UPDATED IN GO BACK ROUTINE)
1212
RETURN WITH UPDATED PQ FOR MN

Fig. 12

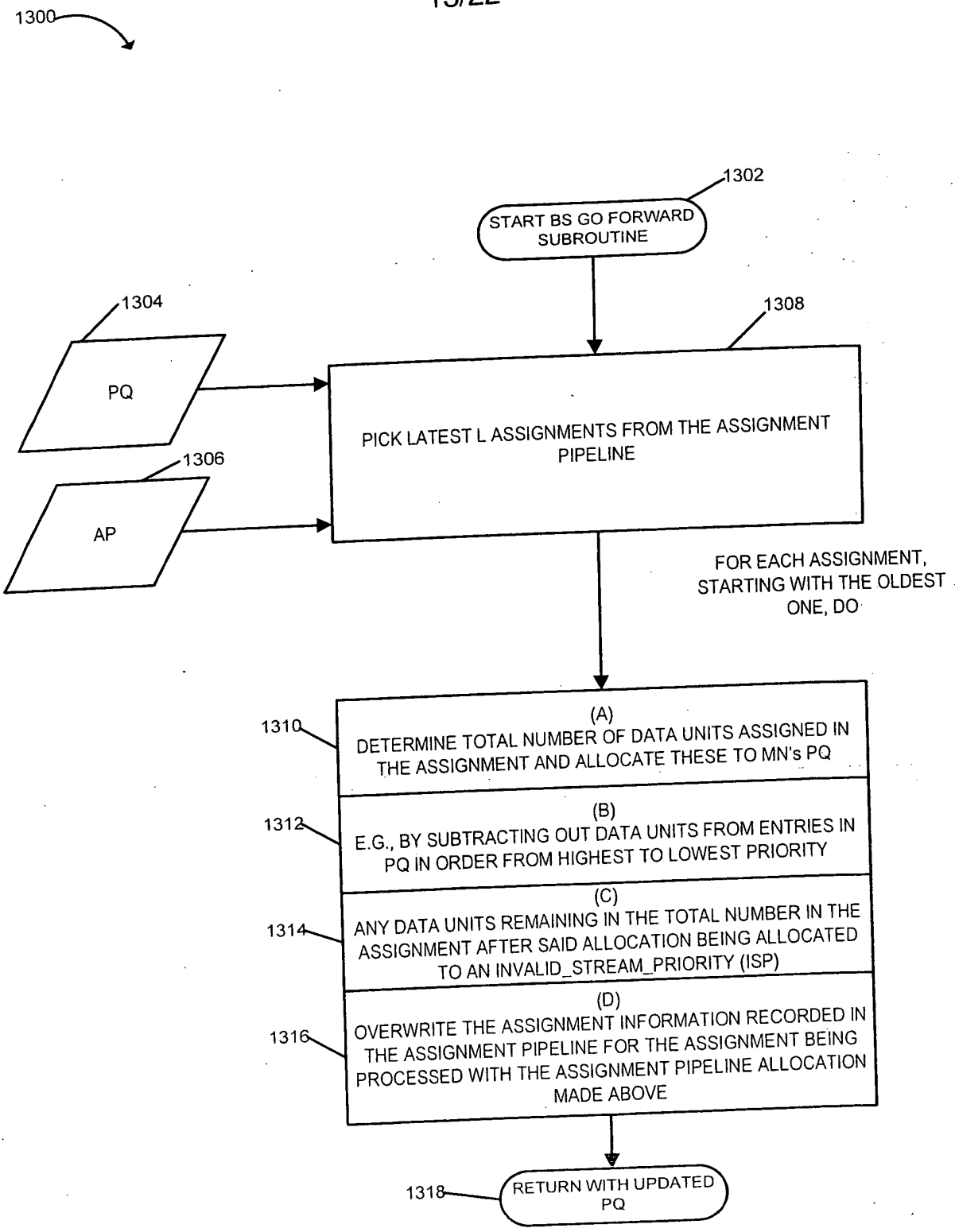

1300
1302
START BS GO FORWARD SUBROUTINE
1304
PQ
1306
AP
1308
PICK LATEST L ASSIGNMENTS FROM THE ASSIGNMENT PIPELINE
FOR EACH ASSIGNMENT, STARTING WITH THE OLDEST ONE, DO
1310
(A)
DETERMINE TOTAL NUMBER OF DATA UNITS ASSIGNED IN THE ASSIGNMENT AND ALLOCATE THESE TO MN's PQ
1312
(B)
E.G., BY SUBTRACTING OUT DATA UNITS FROM ENTRIES IN PQ IN ORDER FROM HIGHEST TO LOWEST PRIORITY
1314
(C)
ANY DATA UNITS REMAINING IN THE TOTAL NUMBER IN THE ASSIGNMENT AFTER SAID ALLOCATION BEING ALLOCATED TO AN INVALID_STREAM_PRIORITY (ISP)
1316
(D)
OVERWRITE THE ASSIGNMENT INFORMATION RECORDED IN THE ASSIGNMENT PIPELINE FOR THE ASSIGNMENT BEING PROCESSED WITH THE ASSIGNMENT PIPELINE ALLOCATION MADE ABOVE
1318
RETURN WITH UPDATED PQ

Fig 13

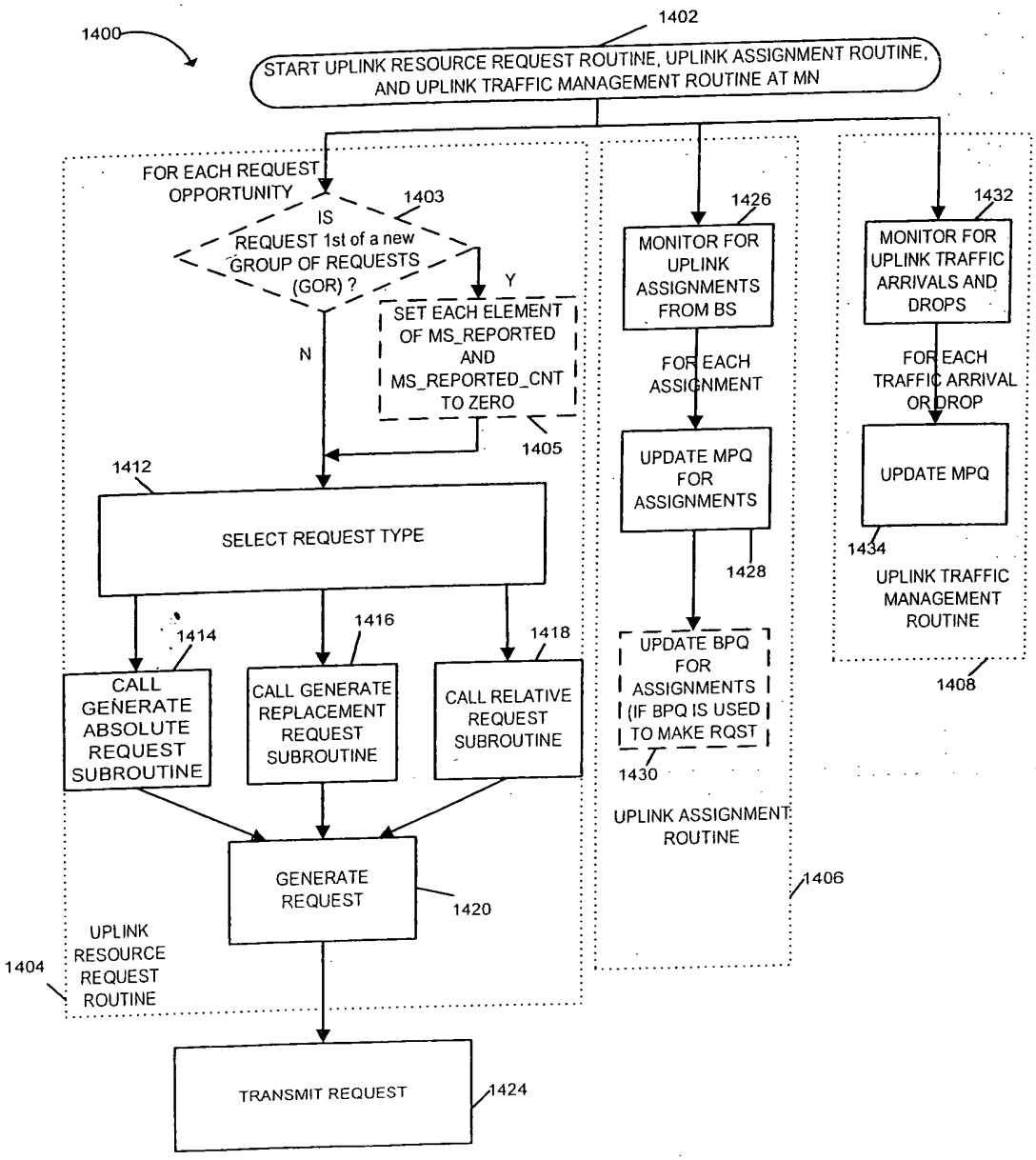

1400
1402
START UPLINK RESOURCE REQUEST ROUTINE, UPLINK ASSIGNMENT ROUTINE, AND UPLINK TRAFFIC MANAGEMENT ROUTINE AT MN
FOR EACH REQUEST OPPORTUNITY
1403
IS REQUEST 1st of a new GROUP OF REQUESTS (GOR) ?
Y
N
SET EACH ELEMENT OF MS_REPORTED AND MS_REPORTED_CNT TO ZERO
1405
1412
SELECT REQUEST TYPE
1414
1416
1418
CALL GENERATE ABSOLUTE REQUEST SUBROUTINE
CALL GENERATE REPLACEMENT REQUEST SUBROUTINE
CALL RELATIVE REQUEST SUBROUTINE
GENERATE REQUEST
1420
UPLINK RESOURCE REQUEST ROUTINE
1404
TRANSMIT REQUEST
1424
1426
MONITOR FOR UPLINK ASSIGNMENTS FROM BS
FOR EACH ASSIGNMENT
UPDATE MPQ FOR ASSIGNMENTS
1428
UPDATE BPQ FOR ASSIGNMENTS (IF BPQ IS USED TO MAKE RQST
1430
UPLINK ASSIGNMENT ROUTINE
1406
1432
MONITOR FOR UPLINK TRAFFIC ARRIVALS AND DROPS
FOR EACH TRAFFIC ARRIVAL OR DROP
UPDATE MPQ
1434
UPLINK TRAFFIC MANAGEMENT ROUTINE
1408

Fig. 14

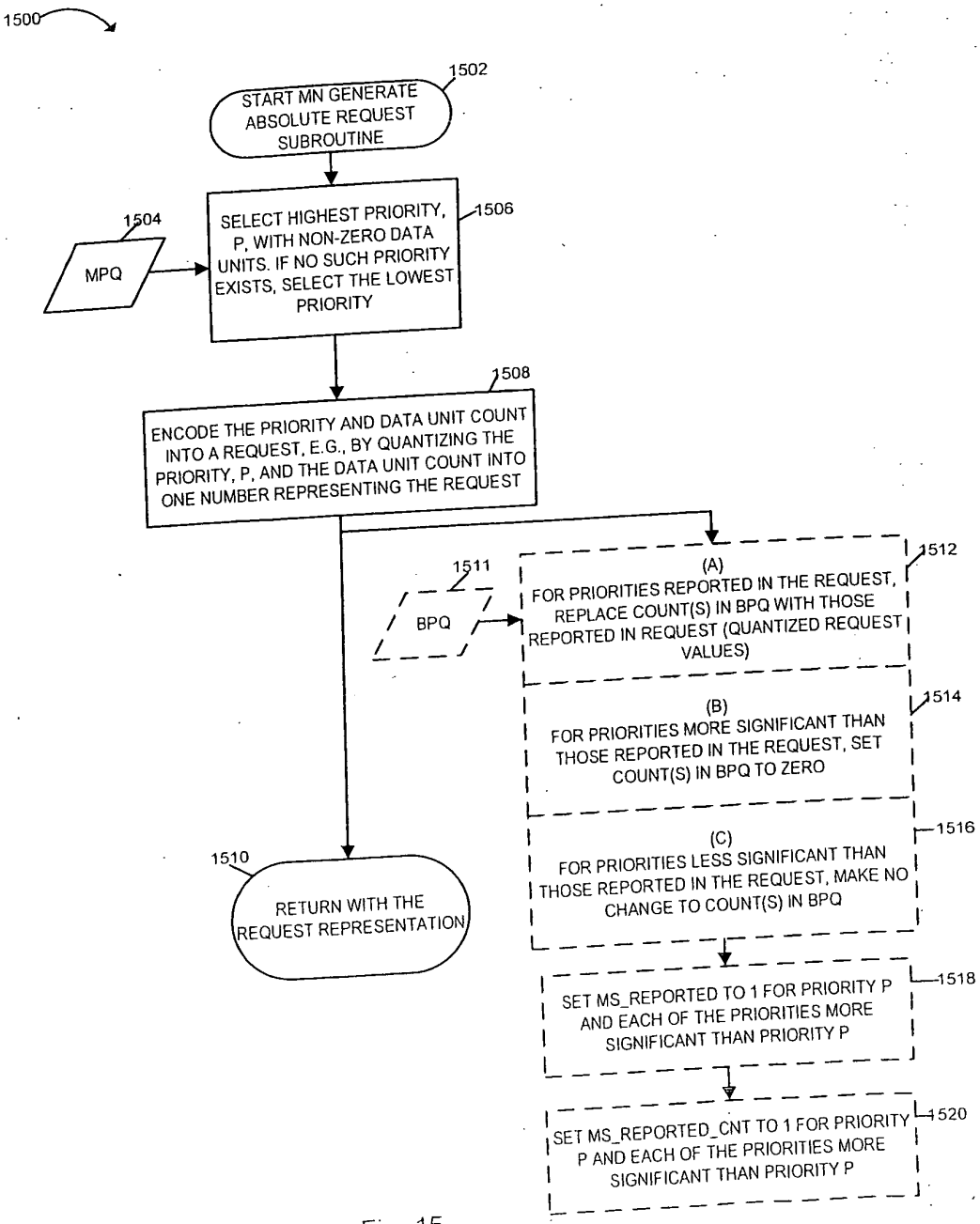

1500
START MN GENERATE ABSOLUTE REQUEST SUBROUTINE
1502
MPQ
1504
SELECT HIGHEST PRIORITY, P, WITH NON-ZERO DATA UNITS. IF NO SUCH PRIORITY EXISTS, SELECT THE LOWEST PRIORITY
1506
ENCODE THE PRIORITY AND DATA UNIT COUNT INTO A REQUEST, E.G., BY QUANTIZING THE PRIORITY, P, AND THE DATA UNIT COUNT INTO ONE NUMBER REPRESENTING THE REQUEST
1508
BPQ
1511
(A) FOR PRIORITIES REPORTED IN THE REQUEST, REPLACE COUNT(S) IN BPQ WITH THOSE REPORTED IN REQUEST (QUANTIZED REQUEST VALUES)
1512
(B) FOR PRIORITIES MORE SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST, SET COUNT(S) IN BPQ TO ZERO
1514
(C) FOR PRIORITIES LESS SIGNIFICANT THAN THOSE REPORTED IN THE REQUEST, MAKE NO CHANGE TO COUNT(S) IN BPQ
1516
RETURN WITH THE REQUEST REPRESENTATION
1510
SET MS_REPORTED TO 1 FOR PRIORITY P AND EACH OF THE PRIORITIES MORE SIGNIFICANT THAN PRIORITY P
1518
SET MS_REPORTED_CNT TO 1 FOR PRIORITY P AND EACH OF THE PRIORITIES MORE SIGNIFICANT THAN PRIORITY P
1520

Fig. 15

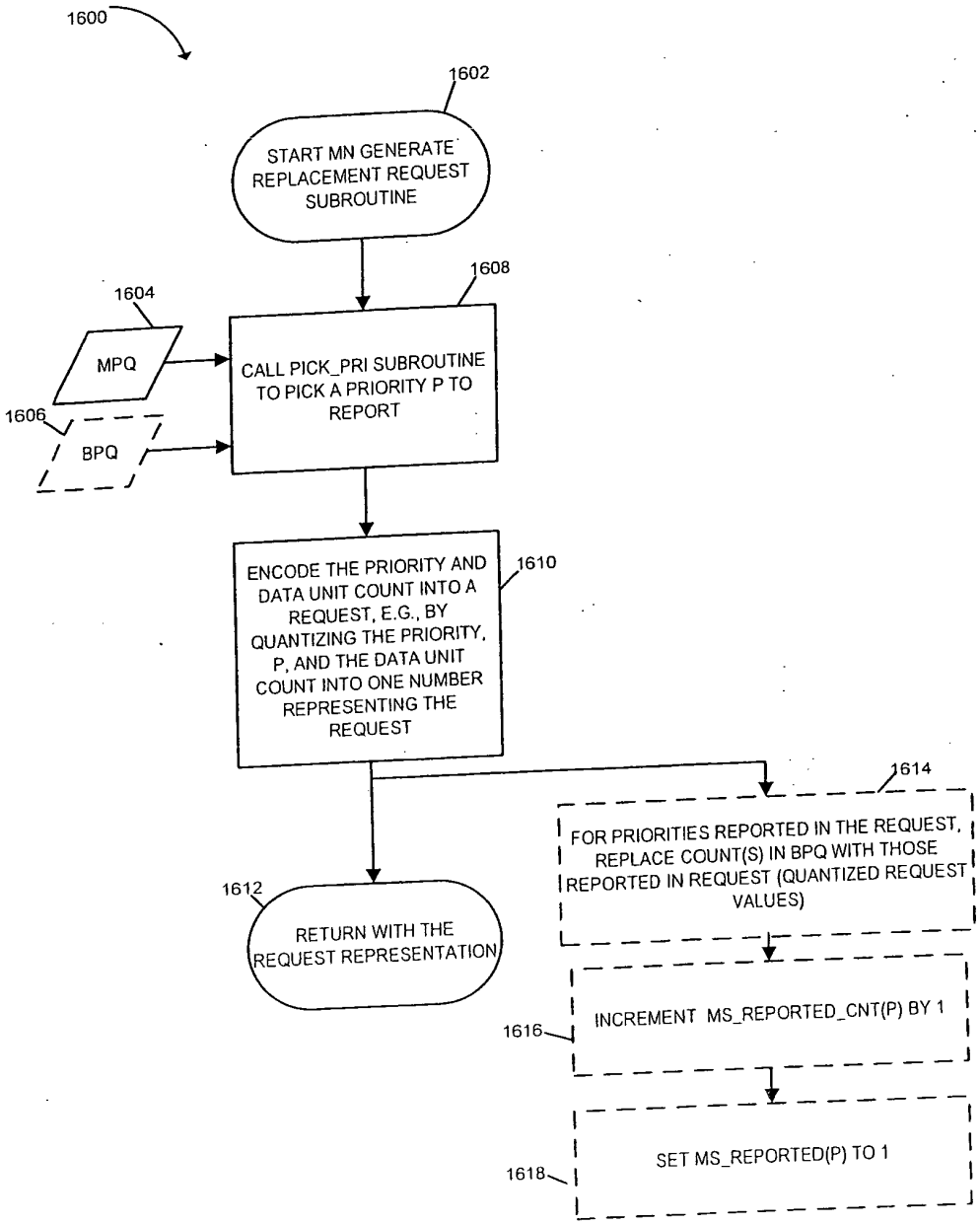

1600
1602
START MN GENERATE REPLACEMENT REQUEST SUBROUTINE
1604
MPQ
1606
BPQ
1608
CALL PICK_PRI SUBROUTINE TO PICK A PRIORITY P TO REPORT
ENCODE THE PRIORITY AND DATA UNIT COUNT INTO A REQUEST, E.G., BY QUANTIZING THE PRIORITY, P, AND THE DATA UNIT COUNT INTO ONE NUMBER REPRESENTING THE REQUEST
1610
1612
RETURN WITH THE REQUEST REPRESENTATION
1614
FOR PRIORITIES REPORTED IN THE REQUEST, REPLACE COUNT(S) IN BPQ WITH THOSE REPORTED IN REQUEST (QUANTIZED REQUEST VALUES)
1616
INCREMENT MS_REPORTED_CNT(P) BY 1
1618
SET MS_REPORTED(P) TO 1

Fig. 16

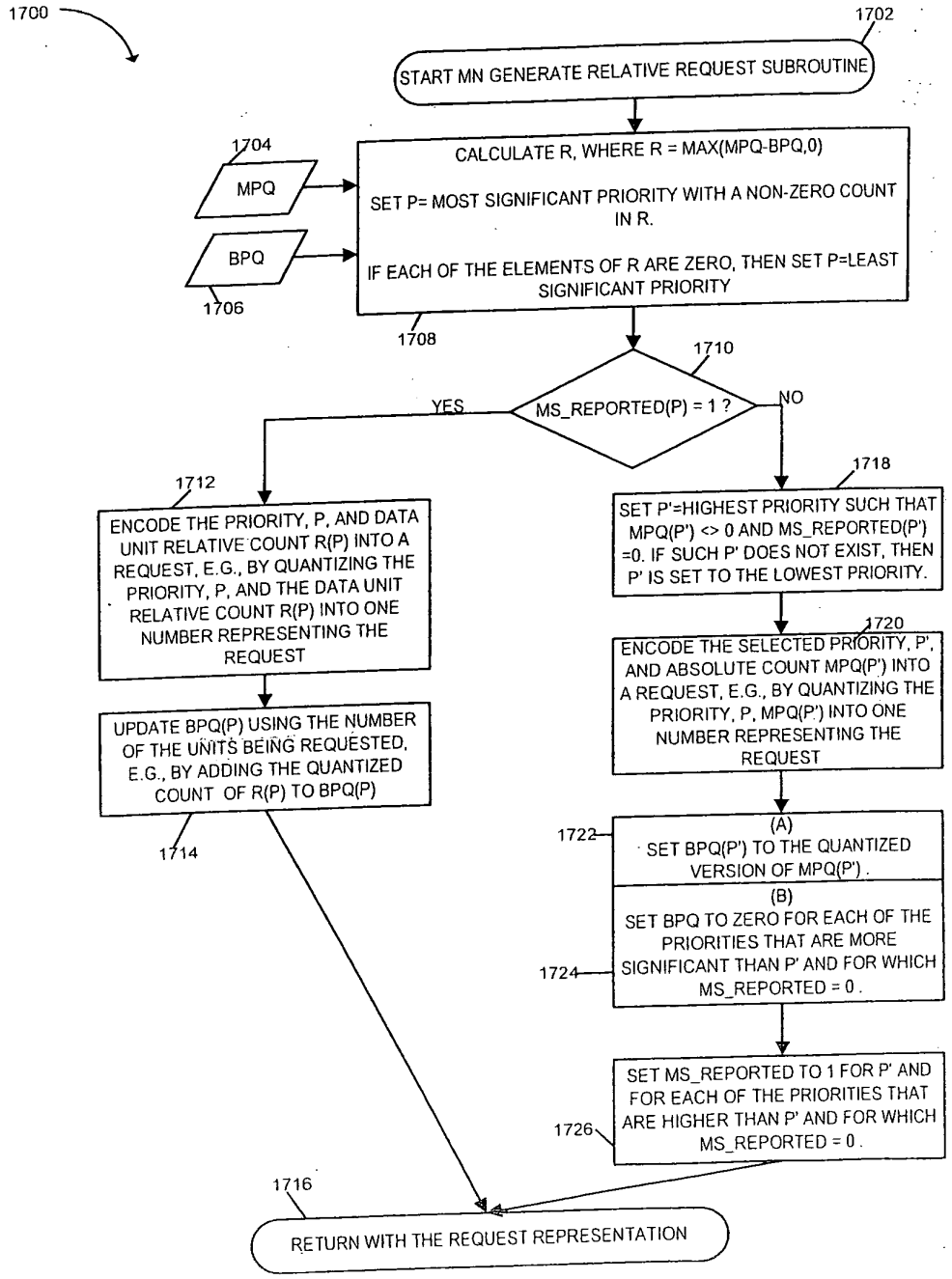

1700
1702
START MN GENERATE RELATIVE REQUEST SUBROUTINE
1704
MPQ
BPQ
1706
CALCULATE R, WHERE R = MAX(MPQ-BPQ,0)
SET P= MOST SIGNIFICANT PRIORITY WITH A NON-ZERO COUNT IN R.
IF EACH OF THE ELEMENTS OF R ARE ZERO, THEN SET P=LEAST SIGNIFICANT PRIORITY
1708
1710
MS_REPORTED(P) = 1 ?
YES
NO
1712
ENCODE THE PRIORITY, P, AND DATA UNIT RELATIVE COUNT R(P) INTO A REQUEST, E.G., BY QUANTIZING THE PRIORITY, P, AND THE DATA UNIT RELATIVE COUNT R(P) INTO ONE NUMBER REPRESENTING THE REQUEST
UPDATE BPQ(P) USING THE NUMBER OF THE UNITS BEING REQUESTED, E.G., BY ADDING THE QUANTIZED COUNT OF R(P) TO BPQ(P)
1714
1718
SET P'=HIGHEST PRIORITY SUCH THAT MPQ(P') <> 0 AND MS_REPORTED(P') =0. IF SUCH P' DOES NOT EXIST, THEN P' IS SET TO THE LOWEST PRIORITY.
1720
ENCODE THE SELECTED PRIORITY, P', AND ABSOLUTE COUNT MPQ(P') INTO A REQUEST, E.G., BY QUANTIZING THE PRIORITY, P, MPQ(P') INTO ONE NUMBER REPRESENTING THE REQUEST
1722
(A)
SET BPQ(P') TO THE QUANTIZED VERSION OF MPQ(P') .
(B)
SET BPQ TO ZERO FOR EACH OF THE PRIORITIES THAT ARE MORE SIGNIFICANT THAN P' AND FOR WHICH MS_REPORTED = 0 .
1724
SET MS_REPORTED TO 1 FOR P' AND FOR EACH OF THE PRIORITIES THAT ARE HIGHER THAN P' AND FOR WHICH MS_REPORTED = 0 .
1726
1716
RETURN WITH THE REQUEST REPRESENTATION

Fig. 17

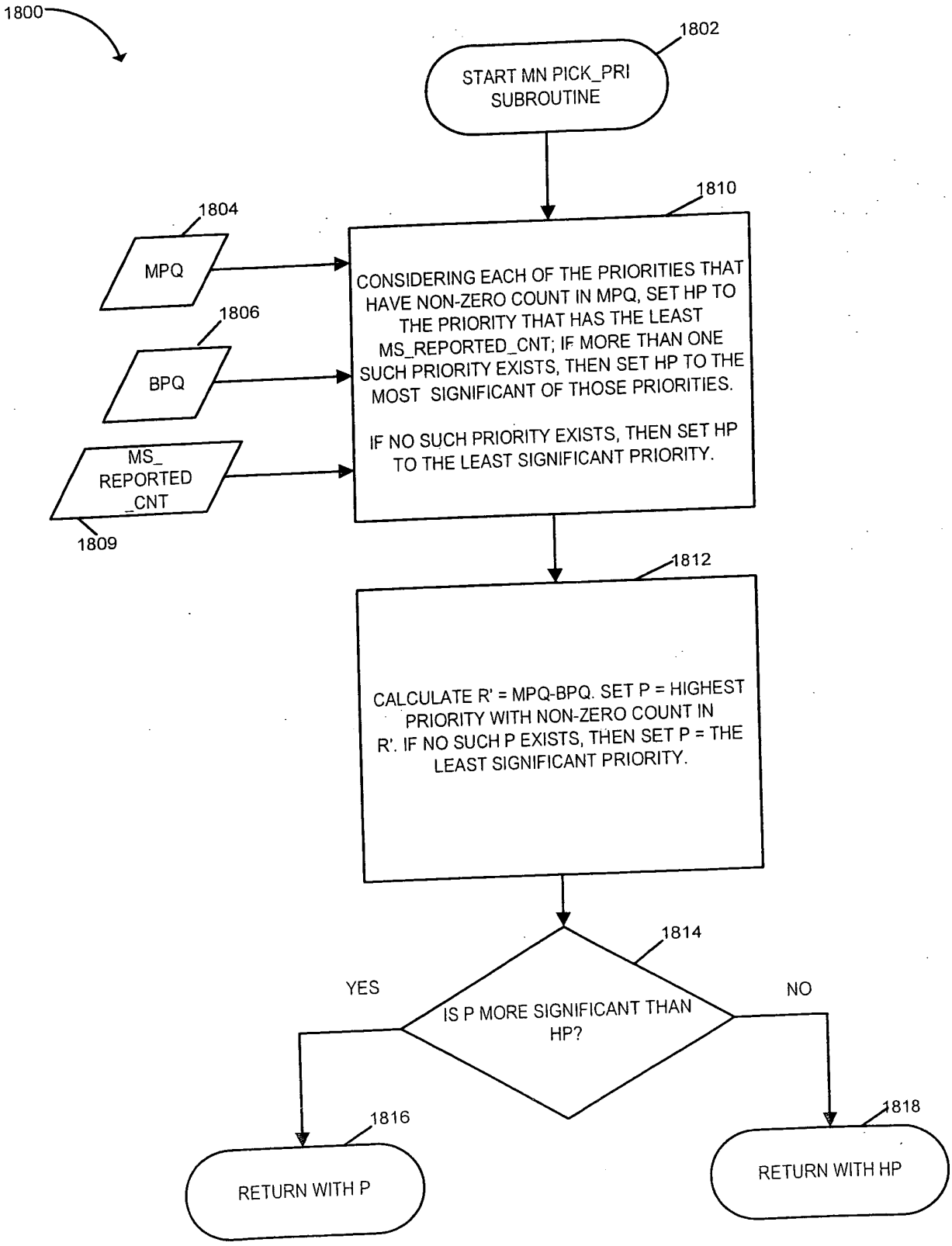

1800
1802
START MN PICK_PRI SUBROUTINE
1804
MPQ
1806
BPQ
MS_ REPORTED _CNT
1809
1810
CONSIDERING EACH OF THE PRIORITIES THAT HAVE NON-ZERO COUNT IN MPQ, SET HP TO THE PRIORITY THAT HAS THE LEAST MS_REPORTED_CNT; IF MORE THAN ONE SUCH PRIORITY EXISTS, THEN SET HP TO THE MOST SIGNIFICANT OF THOSE PRIORITIES.
IF NO SUCH PRIORITY EXISTS, THEN SET HP TO THE LEAST SIGNIFICANT PRIORITY.
1812
CALCULATE R' = MPQ-BPQ. SET P = HIGHEST PRIORITY WITH NON-ZERO COUNT IN R'. IF NO SUCH P EXISTS, THEN SET P = THE LEAST SIGNIFICANT PRIORITY.
1814
IS P MORE SIGNIFICANT THAN HP?
YES
NO
1816
RETURN WITH P
1818
RETURN WITH HP

Fig. 18

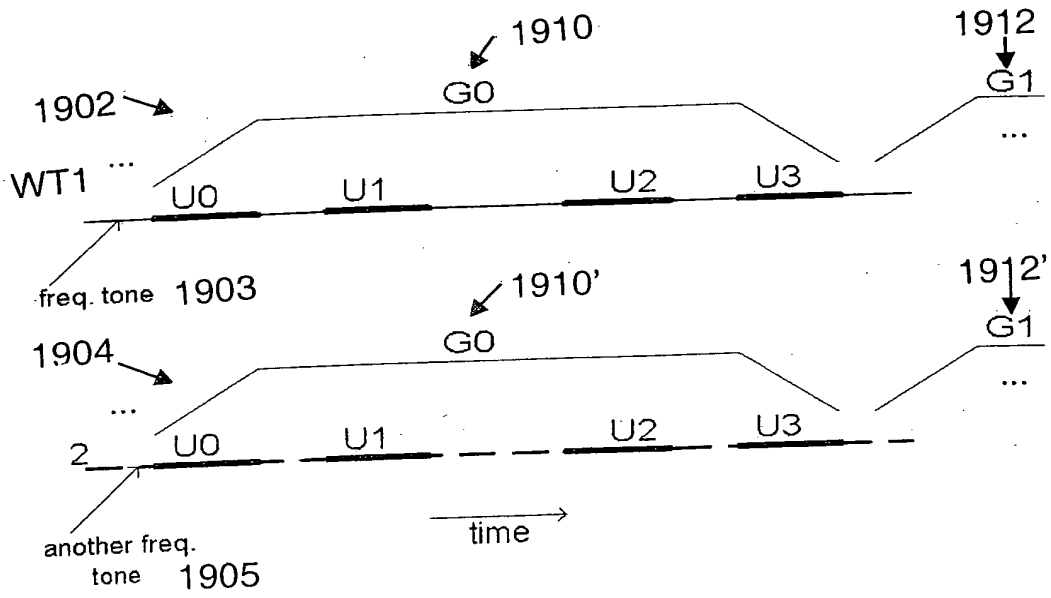

1910
1912
G0
G1
1902
WT1
U0
U1
U2
U3
freq. tone 1903
1910'
1912'
G0
G1
1904
2
U0
U1
U2
U3
time
another freq. tone 1905

Fig. 19

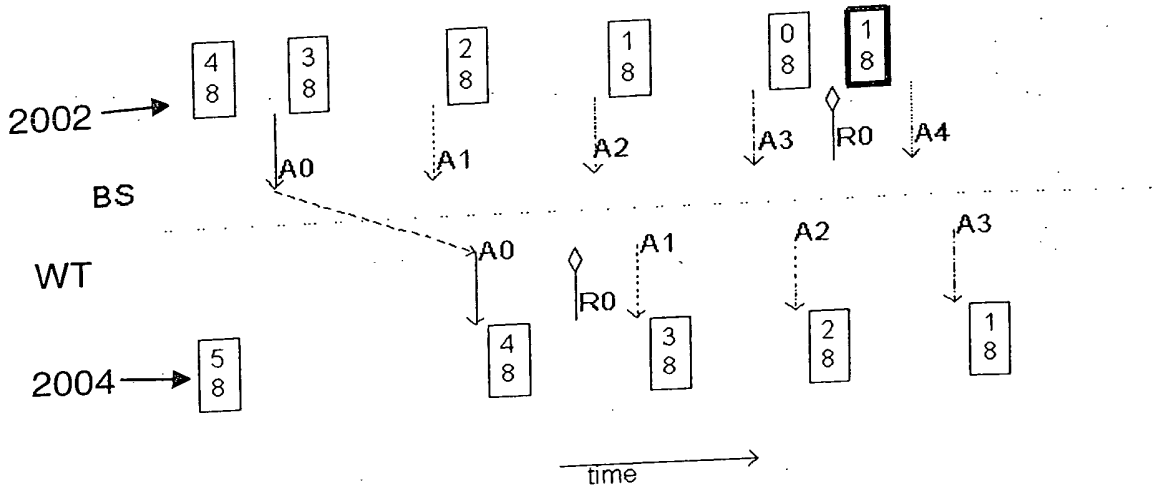

2002
BS
WT
2004
A0
A1
A2
A3
R0
A4
time

Fig. 20

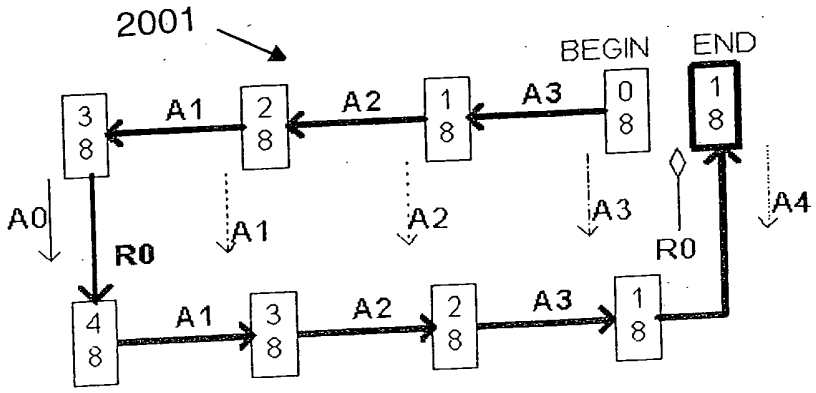

2001
BEGIN
END
A0
A1
A2
A3
A4
R0

Fig. 21

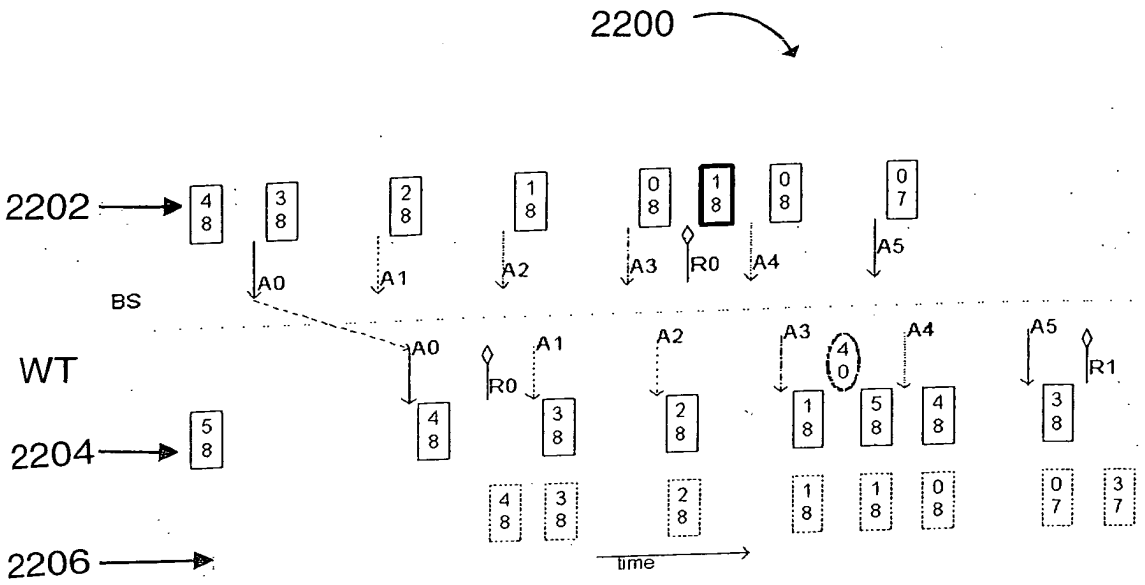


Fig. 22

2300

2301 5 bit request	2302 Priority	2304 n at MS	2306 n at BS	Comment
0	X	X	X	Not used for uplink request
1	X	X	X	Not used for uplink request
2	X	X	X	Not used for uplink request
3	3	n=0	n=0	
4	1	n=1	n=1	
5	1	n=2	n=2	
6	1	n=3	n=3	
7	1	n>=4	n=4	
8	0	n=1	n=1	
9	0	n=2	n=1	
10	0	n=3	n=1	
11	0	n=4	n=1	
12	0	n=5	n=1	
13	0	n=6	n=1	
14	0	n=7	n=1	
15	0	n>=8	n=1	
16	2	thre[0] <=n < thre[1]	n=thre[0]	
17	2	thre[1] <=n < thre[2]	n=thre[1]	
18	2	thre[2] <=n < thre[3]	n=thre[2]	
19	2	thre[3] <=n < thre[4]	n=thre[3]	
20	2	thre[4] <=n < thre[5]	n=thre[4]	
21	2	thre[5] <=n < thre[6]	n=thre[5]	
22	2	thre[6] <=n < thre[7]	n=thre[6]	
23	2	thre[7] <=n	n=thre[7]	
24	3	thre[0] <=n < thre[1]	n=thre[0]	
25	3	thre[1] <=n < thre[2]	n=thre[1]	
26	3	thre[2] <=n < thre[3]	n=thre[2]	
27	3	thre[3] <=n < thre[4]	n=thre[3]	
28	3	thre[4] <=n < thre[5]	n=thre[4]	
29	3	thre[5] <=n < thre[6]	n=thre[5]	
30	3	thre[6] <=n < thre[7]	n=thre[6]	
31	3	thre[7] <=n	n=thre[7]	

FIG. 23